

TRANSFORM-DOMAIN SAMPLE-BY-SAMPLE  
DECISION FEEDBACK EQUALIZER

5

The present invention is directed to methods and apparatuses for processing received digital signals in a digital communications system, and more particularly to a method and apparatus for processing received digital signals in a digital communications system using packet based signals in the presence of noise and inter-symbol interference.

Adaptive equalizers are often used in digital communication systems to mitigate inter-symbol interference caused by multi-path. Among the many variants of adaptive equalizers, Least Mean Squares (LMS) type decision feedback equalizers are the most frequently used. For emerging applications where the channel is dynamic or requiring fast convergence speed, the traditional LMS-type equalizers often exhibit inadequate performance.

The convergence speed of traditional time-domain LMS type adaptive equalizers depends on the ratio of the maximum to the minimum eigenvalues of the autocorrelation matrix of the input. Filters having inputs with a wide eigenvalue spread often take longer to converge than filters with white noise inputs.

As a remedy to this problem, transform domain equalizers were developed. These equalizers are based on orthogonalization of the input signals, which are often referred to as frequency-domain adaptive filters. Such orthogonalization techniques have been used in the context of linear (FIR) adaptive filters. Simulations have shown that such equalizers have better convergence properties compared to the counterpart time-domain LMS algorithms. Unfortunately, linear equalizers perform very poorly if the channel spectrum contains dip nulls or the inverse of the channel has strong samples outside the range of the linear equalizer. As a result, they suffer from noise-enhancement or lack adequate numbers of taps. Inter symbol interference (ISI) due to multipath can be effectively rejected using non-linear equalizers, such as Decision Feedback Equalizers (DFEs).

Non-linear equalization techniques, such as Decision Feedback Equalizers, exhibit superior performance when compared on the basis of identical numbers of taps and tap-adaptation algorithms.

The inherent performance advantage of DFEs to combat severe multipath interference makes them attractive for practical channel equalization applications. Nevertheless, DFEs are often used in conjunction with LMS-type algorithms for tap adaptations. As a result, the convergence speed of LMS type or blind equalizers is still 5 dependent on the eigenvalue spread of the input. As an alternative, different techniques have been proposed, such as Recursive Least Squares (RLS), etc. However, implementation complexity often precludes the use of such tap adaptation algorithms in practical applications.

The present invention is therefore directed to the problem of developing a method 10 and apparatus for increasing the convergence speed of a digital channel equalizer without unduly increasing the implementation complexity.

The present invention solves these and other problems by providing an adaptive transform-domain decision feedback equalizer with a convergence speed that is faster than 15 the traditional counterpart at a modest increase in computational complexity.

According to one aspect of the present invention, a method for performing equalization on an input signal in a receiver creates multiple delayed samples of the input signal and orthogonally transforms each of the delayed input samples before weighting them using transformed adaptive coefficients. The weighted orthogonally-transformed 20 delayed input samples are summed along with a feedback signal and the result is output as the equalizer output signal.

According to another aspect of the present invention, the feedback signal is formed from delayed samples of a receiver decision signal, which are orthogonally transformed, then weighted using transformed adaptive coefficients, and finally summed and fed back as 25 the feedback signal.

According to another aspect of the present invention, the feedback signal is formed from delayed samples of a receiver decision signal, which are weighted using adaptive coefficients, and finally summed and fed back as the feedback signal.

An exemplary embodiment of the equalizer herein is particularly suitable for 30 applications with small delay dispersions, such as home networks or LANs.

FIG 1 depicts a conventional receiver with an equalizer.

FIG 2 depicts a block diagram of an exemplary embodiment of an apparatus for performing a Transform-Domain Decision Feedback Equalizer (TDDFE) according to one aspect of the present invention.

5       FIG 3 depicts a block diagram of an exemplary embodiment of an apparatus for performing a Hybrid DFE (HDFE) according to another aspect of the present invention.

FIG 4 depicts an exemplary embodiment of a method for computing a transform of a sequence in a recursive manner according to yet another aspect of the present invention.

10      FIG 5 depicts a graph of simulation results of the embodiment of FIG 1 for a paper channel,  $\mu_1=20$ ,  $\mu_2=16$ , LMS (factor 32), averaged over 500 symbols.

It is worthy to note that any reference herein to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention.

15      The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment.

Turning to FIG 1, shown therein is an exemplary embodiment of a digital receiver 10 according to one aspect of the present invention. The receiver 10 includes an antenna 11, an analog front end (e.g., a filter, tuner, etc.) 12, and analog-to-digital converter (ADC) 20 13, a timing/carrier recovery circuit 14, an adaptive equalizer 15 having its own processor 18, a phase-corrector 16 and a receiver decision device (such as a forward error corrector or trellis decoder) 17. 23.

In this exemplary embodiment, the adaptive equalizer (e.g., 20, 30) is also coupled to the receiver decision device (17) and includes a processor (18) to: create delayed 25 versions of the digital signal; orthogonally transform the delayed versions and weight them using transformed adaptive coefficients; sum the weighted and orthogonally-transformed delayed versions of the digital signal along with a feedback signal to create an equalized output signal. Moreover, the processor adaptively updates the transformed adaptive coefficients based on decisions made in the receiver decision device using prior versions of 30 an equalized output signal. This updating is performed in the conventional manner, except that the adaptive coefficients are orthogonally transformed as set forth below.

The present invention concerns the equalization stage of a receiver. The present invention allows decision-feedback equalizers to converge faster than conventional equalizers. Fast convergence is essential in bi-directional packet based digital communications systems, such as wireless Local Area Networks (LANs). Thus, the 5 present invention makes possible receivers that can operate in places with high levels of multipath or that can switch faster between channels/cells. The embodiments herein can be employed in any digital communications system, such as a wireless LAN, which requires channel equalization.

An exemplary embodiment of the present invention provides a transform domain 10 decision feedback equalizer technique. Further, the description herein includes performance evaluations using simulations. The exemplary embodiment exhibits superior performance compared to the traditional LMS type DFEs. From an implementation point of view, this technique is well suited for applications requiring small numbers of taps.

The Transform-Domain DFE, termed herein TDDFE, is based on applying 15 orthogonal transformation of the inputs of both the forward and feedback sections. The orthogonal transform can be a Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), or another similar transform. The taps of the TDDFE are updated in the inverse-transform domain using orthogonalization techniques. In order to proceed further, we revise the conventional time-domain LMS-DFE relationships:

$$\begin{aligned}
 20 \quad y(n) &= \mathbf{x}(n) \cdot \mathbf{c}^T(n) + \mathbf{b}(n-1) \cdot \mathbf{f}^T(n) \\
 e(n) &= d(n) - y(n) \\
 \mathbf{c}(n+1) &= \mathbf{c}(n) + \mu e(n) \mathbf{x}(n)^* \\
 \mathbf{f}(n+1) &= \mathbf{f}(n) + \mu e(n) \mathbf{b}(n-1)^* \\
 b(n) &= dd[y(n)] \tag{1}
 \end{aligned}$$

25 where  $y(n)$  is the output of the equalizer,  $\mathbf{x}(n)=\{\mathbf{x}(n), \mathbf{x}(n+1), \dots, \mathbf{x}(n-M+1)\}$  is a vector consisting of the samples of the equalizer input,  $\mathbf{b}(n)=\{\mathbf{b}(n), \mathbf{b}(n+1), \dots, \mathbf{b}(n-N+1)\}$  is a vector consisting of the input samples  $b(n)$  of the feedback section,  $\mathbf{c}(n)$  is an M-length vector consisting of the coefficients of the feed-forward section of the equalizer,  $\mathbf{f}(n)$  is an N-length vector consisting of the coefficients of the feedback section of the equalizer,  $d(n)$  30 is a reference signal or a locally generated decision term,  $e(n)$  is the error term,  $\mu$  is the adaptation step size,  $dd[y(n)]$  is the decision device, '\*' denotes complex conjugate and ' $T$ ',

denotes transpose operation. We define the orthogonal transform operation in a square matrix format  $T_1$  and  $T_2$  where  $T_1$  is an  $M \times M$  square matrix and  $T_2$  is an  $N \times N$  square matrix. The inverse of these matrices represent the inverse transform, i.e.,

$T_1 \cdot T_1^{-1} = T_2 \cdot T_2^{-1} = I$ , where  $I$  is the identity matrix. Using this property of the matrices,

5 equation (1) can be described as:

$$y(n) = \mathbf{x}(n)T_1 \cdot T_1^{-1}\mathbf{c}^T(n) + \mathbf{b}(n-1)T_2 \cdot T_2^{-1}\mathbf{f}^T(n) \quad (2)$$

Defining the transformed variables  $\chi(n) = \mathbf{x}(n) \cdot T_1$ ,  $\xi(n) = \mathbf{c}(n)T_1^{-1T}$ ,  $\beta(n) = \mathbf{b}(n)T_2$ , and  $\nu(n) = \mathbf{f}(n)T_2^{-1T}$ , the above equation can be described as:

$$y(n) = \chi(n)\xi^T(n) + \beta(n-1)\nu^T(n) \quad (3)$$

10 Equation (3) describes the input output relationship of the TDDFE equalizer where the output is computed using the transformed variables. By multiplying both sides of the tap-adaptation equations in (1) with the transform matrices  $T_1$  and  $T_2$  and considering only the transform operation where  $T_2 = T_2^*$ , we easily find:

$$\xi(n+1) = \xi(n) + \mu e(n)\chi^*(n)$$

15  $\nu(n+1) = \nu(n) + \mu e(n)\beta^*(n) \quad (4)$

The above equations are just an alternative way of describing the time-domain LMS-type equalizer in the transform domain. As a result, there is no reason to expect that the performance of this equalizer will be different from that of the time-domain counterpart.

Nevertheless, these equations provide a simple means by which orthogonalization of the

20 input can be achieved to obtain an equalizer that converges faster and exhibits better tracking behavior. Orthogonalization is achieved by measuring the average power of the transformed inputs and using these in the tap-adaptation equations (4). Defining the average values as:

$$\begin{aligned} \Gamma_x(n+1) &= \lambda \Gamma_x(n) + |\chi(n)|^2 \\ \Gamma_b(n+1) &= \lambda \Gamma_b(n) + |\beta(n)|^2 \end{aligned} \quad (5)$$

where  $|\cdot|^2$  is an element-wise magnitude operator,  $\lambda$  is a positive constant, and

$\Gamma_x(n)$  and  $\Gamma_b(n)$  are the respective average values, the tap-update equations are then modified to:

$$\xi(n+1) = \xi(n) + \mu e(n)\chi^*(n) \cdot / \Gamma_x(n)$$

$$\nu(n+1) = \nu(n) + \mu e(n) \beta^*(n) \cdot / \Gamma_b(n) \quad (6)$$

- where the operation ‘./’ is an element-wise vector division. The gradient terms in (6) consist of almost uncorrelated variables due to the orthogonal transform operation. As a result, each frequency bin is weighted by variables that are not dependent on the other  
 5 variables. This is similar to having a time-varying adaptation constant for each tap of the equalizer. Since the tap-adaptation is done using uncorrelated variables, it is natural to expect that the convergence speed of this equalizer is relatively insensitive to the eigenvalue spread and that it converges faster than the traditional time-domain LMS equalizer. As pointed out below, this algorithm is in fact an approximate RLS algorithm.  
 10 The vectors  $\Gamma_x$  and  $\Gamma_b$  are the diagonal elements of the autocorrelation matrix. As a result of this type of RLS type approximation, it shares the behavior of the standard RLS algorithm, but at a reduced computational complexity.

An exemplary embodiment 20 of the top-level architecture of this transform-domain DFE is shown in FIG 2. The input to the equalizer is fed into N taps 2-1 through  
 15 2-N. Each of the outputs from the taps is fed into transform 24.

Transform 24 performs an orthogonal transform on the outputs of the taps 2-1 through 2-N. Various types of orthogonal transforms can be employed, such as a Fast Fourier Transform (FFT) or a Discrete Cosine Transform (DCT), to name only a few. FIG 4 depicts an exemplary embodiment of the transform 24.

The transform outputs, of which there are N in number, are then weighted by the  $\zeta(n)$  coefficients, which are the transformed filter coefficients  $c(n)$ , which are the original filter coefficients used in a conventional Decision Feedback Equalizer.

The weighted and transformed tap outputs (i.e., the inputs to summer 25) are then summed at summer 25 and the result is output to decision device 26. The output of the  
 25 decision device 26 is also fed through N taps 7-1 through 7-N, the outputs of which are fed through another transform 28.

Transform 28 performs an orthogonal transform on the outputs of the taps 7-1 through 7-N, similar to that employed in transform 24. However, a different type of orthogonal transform can be employed than that used in transform 24. The transform  
 30 shown in FIG 4 could also be applied as transform 28.

The outputs of the transform 28, of which there are N in number, are then weighted by the transformed filter coefficients  $\nu(n)$ , the results of which weighting are then summed at summer 27 and fed back into summer 25.. The equalizer output is then output from the summer 25 for processing by the receiver decision device 26, which outputs a  
5 receiver decision signal to taps 7-1 through 7-N as described above.

The orthogonal transform converts its input to an uncorrelated variable. As a result, faster convergence is expected if diagonalized tap adaptation is employed. However, if the input itself is uncorrelated, then there is no adequate reason to expect performance  
10 advantages of the transform domain operations. This rationale can be used to simplify the operations needed for the transform domain DFE.

Decision-Feedback Equalizers exhibit superior performance when the input samples of the feedback filter are correct, i.e., these samples equal the transmitted samples. If the decision device produces correct decisions, then the input of the feedback filter is  
15 naturally uncorrelated. This is based on the assumption that the transmitted sequence is also uncorrelated. Under these assumptions, there is no reason to expect performance advantage by operating the feedback part of the DFE in the transform domain. In order to save computation due to the transform, the feedback part can operate in the pure time-domain mode. This mode of operation is hereinafter referred to as Hybrid Decision-  
20 Feedback Equalization (HDFE). The input output and tap-update relationship of the HDFE can be described by:

$$\begin{aligned} y(n) &= \chi(n)\xi^T(n) + \mathbf{b}(n-1)\mathbf{f}^T(n) \\ \xi(n+1) &= \xi(n) + \mu_1 e(n)\chi^*(n) \cdot / \Gamma_x(n) \\ \mathbf{f}(n+1) &= \mathbf{f}(n) + \mu_2 e(n)\mathbf{b}(n-1)^* \end{aligned} \quad (7)$$

25 FIG 3 illustrates an exemplary embodiment 30 of the hybrid DFE scheme. The feed forward portion of the circuit 30 remains unchanged from the embodiment 20 shown in FIG 2. The input to the equalizer 30 is fed into N taps 2-1 through 2-N. Each of the outputs from the taps 2-1 through 2-N is fed into transform 24.

Transform 24 performs an orthogonal transform on the outputs of the taps 2-1  
30 through 2-N. As in the above embodiment, various types of orthogonal transforms can be

employed, such as a Fast Fourier Transform (FFT) or a Discrete Cosine Transform (DCT), to name only a few. The transform shown in FIG 4 could be employed here as well.

The transform outputs, of which there are N in number, are then weighted by the  $\zeta(n)$  coefficients, which are the transformed filter coefficients  $c(n)$ , which are the original  
5 filter coefficients used in a conventional Decision Feedback Equalizer.

The weighted and transformed tap outputs (i.e., the inputs to summer 25) are then summed at summer 25 and the result is output to decision device 26. The output of the decision device 26 is also fed through N taps 7-1 through 7-N. This is where the two embodiments differ.

10 The tap outputs, of which there are N in number, are then weighted by the feedback filter coefficients  $f(n)$ , the results of which weighting are then summed at summer 27 and fed back into summer 25 as error signals. The equalizer output is then output from the summer 25 for processing by the receiver decision device 26, which outputs a receiver decision signal to taps 7-1 through 7-N as described above.

15 Note that the adaptation step sizes  $\mu$ 's for the transform domain and the time-domain tap adaptation are different. The performance of this equalizer was evaluated using simulation, the results of which are shown in FIG 5.

As mentioned earlier, the feedback decisions will not be correct if the interference (e.g., multipath and noise) is severe and if a simple slicer is used as the decision device. If  
20 the feedback decisions are not correct, the DFE will suffer performance degradation due to these incorrect decisions circulating in the feedback path. This effect is commonly referred to as error-propagation. The error propagation behavior of DFEs is well observed using simulations, but little understood analytically. Rather, researchers opted to employ techniques that reduce error propagation. These techniques are centered on the use of  
25 reliable decision devices. One common method of obtaining reliable decisions is by forming decisions from the results of the forward error correction unit.

As with other stochastic adaptive algorithms, the adaptation step size determines both the learning curve and the steady-state Mean Square Error (MSE). The adaptation step-size of the feedback filter and the forward filter can be set to different values to obtain  
30 better performance. The average values of the power estimate give adequate clues to obtain step sizes. Assuming that the step size of the forward and feedback sections of the time-domain DFE are the same, the step sizes of the TDDFE and the HDFE can be chosen,

such that the average value of the step sizes equal the time-domain step size. For the DFT type transform, this procedure results in the following values of the step sizes.

For the TDDFE:  $\mu_1=pM\mu, \mu_2=pN\mu$  and

For the HDFE:  $\mu_1=pM\mu, \mu_2=\mu$ .

- 5 where  $p$  is the average power of the time-domain input signal, and  $\mu$  is the time-domain step size. The performance of the equalizer is evaluated using these step size relationships.

In order to evaluate the performance of the proposed equalizer, computer simulations were made for both the exemplary embodiment of the equalizer and the time-domain equalizer with equal numbers of taps. A 32-tap forward and a 16-tap feedback 10 equalizer were used. The adaptation constants ( $\mu$ ) for both equalizers were chosen such that the MSE curves of both equalizers exhibit similar convergence performance and steady-state MSE for a channel with a post echo of -6 dB at one sample away from the main path. The simulation results are plotted in FIG 3 for a paper channel,  $\mu_1=20, \mu_2=16$ , LMS (factor 32), averaged over 500 symbols. As these simulation results indicate, the 15 transform-domain equalizer converged faster than the time-domain equalizer, confirming the expectation.

The exemplary embodiment of the transform-domain DFE requires the computation of the transform values on a sample-by-sample basis. As a result, the computational complexity of the exemplary equalizer can be higher than a conventional LMS-type time-domain DFE.

The need for the transform operation every sample precludes using the exemplary embodiment of the equalizer in applications for systems with large delay dispersion, such as terrestrial broadcast applications. Such systems in general require long filters to combat the interference effects of the channel with long delay dispersions. For such applications, a 25 block frequency domain equalizer will be preferable to reduce the computational complexity.

However, the computational complexity of the exemplary embodiment can be reduced to such an extent that the remaining added complexity is offset by the performance advantages. The resulting equalizer is most suitable for small delay dispersion systems, 30 such as home networks or LANs.

The computational requirement can be simplified in the following manner. First, the division by the average power can be approximated by a binary shift operation. This is

achieved by approximating the values of  $\Gamma$ 's with the nearest value that is of the form  $2^k$ , where  $k$  is an integer. Second, the sample-by-sample transforms can be computed by using the prior transform values in a recursive manner. In the case of a Discrete-Fourier Transform, this computation can be achieved as follows. The DFT of the sequence  $x_{n-1}$  is 5 described by:

$$X(k, n-1) = \sum_{i=0}^{M-1} x(n-M+i)W^{ik} \quad (8)$$

Similarly, the DFT of  $x_n$  is

$$X(k, n) = \sum_{i=0}^{M-1} x(n-M+1+i)W^{ik} \quad (9)$$

Replacing the index  $1+i$  with  $i$ , we obtain:

$$10 \quad X(k, n) = \sum_{i=1}^M x(n-M+i)W^{(i-1)k} \quad (10)$$

This equation can also be described as:

$$X(k, n) = W^{-k} \left( \sum_{i=0}^{M-1} x(n-M+i)W^{ik} - x(n-M) + x(n)W^{Mk} \right) \quad (11)$$

Since  $W^{Mk}=1$ , and the inner summation equals  $X(k, n-1)$ , we find:

$$X(k, n) = W^{-k} (X(k, n-1) - x(n-M) + x(n)) \quad (12)$$

15 This equation, when applied to each frequency bin, computes the DFT of the sequence in a sample-by-sample basis using the previously computed values in a recursive manner. This form of computation, which is similar to filtering the input sequence using band pass filters, provides computational savings compared to the FFT operation when done on a sample-by-sample basis, i.e., the radix-2 FFT operation requires  $M/2\log_2(M)$  complex 20 multiplications while the exemplary embodiment equalizer requires  $M$  complex multiplications.

FIG 4 illustrates a simplified method to compute the transform of the input sequence in a recursive manner. Shown in FIG 4 is exemplary embodiment 40, in which the equalizer input  $x(n)$  is fed into filter 41, the output of which is coupled to summer 42 which is subtracted from the original input. The output of summer 42 is fed into  $k$  summers 43-1 through 43- $k$ , the other input of which comprise the feedback error signals. The outputs of summers 43-1 through 43- $k$  are fed into  $k$  filters 44-1 through 44- $k$ . The outputs of filters 44-1 through 44- $k$  represent the  $k$  values of  $X(k, n)$ .

The recursive least squares method minimizes the cost function:

$$J(n) = \sum_{i=1}^n \lambda^{n-i} |e(i)|^2 \quad (13)$$

where  $e(n) = d(n) - w(n)u(n)^T$ . The optimum value for the tap weight vector,  $w(n) = [c(n) f(n)]$ , for which the cost function  $J(n)$  attains minimum value is defined by the following

5 set of equations:

$$\Phi(n)w(n) = z(n) \quad (14)$$

where the  $(N+M)$  by  $(N+M)$  correlation matrix is defined by

$$\begin{aligned} \Phi(n) &= \sum_{i=1}^n \lambda^{n-i} u(i)u(i)^H \\ &= \lambda\Phi(n-1) + u(n)u(n)^H \end{aligned} \quad (15)$$

$u(n)$  is the combined input of the forward and feedback path,  $u(n) = [x(n) b(n-1)]$ . The

10  $(M+N)$  by 1 cross correlation vector  $z(n)$  between the filter inputs and the desired response  $d(n)$  is defined by:

$$\begin{aligned} z(n) &= \sum_{i=1}^n \lambda^{n-i} u(i)d(i)^* \\ &= \lambda z(n-1) + u(n)d(n)^* \end{aligned} \quad (16)$$

The recursive equation for updating the least square estimate tap weight vector is then described as:

15  $w(n) = w(n-1) + \Phi(n)^{-1}u(n)^*\epsilon(n) \quad (17)$

where  $\epsilon(n)$  is the a priori estimation error defined by  $\epsilon(n) = d(n) - w(n-1)u(n)^T$ .

Invoking the assumption that the transformed variables  $u(n)$  are uncorrelated (due to the orthogonal transform operation), it can be noticed that the auto-correlation matrix is diagonal. As a result, (17) is identical to (6) except that an adaptation step constant is used

20 in (6).

The hybrid DFE can also be derived in a similar way. If the decision device produces correct decisions, the average values of the transform of the feedback input will be approximately equal at all transform points. As a result, the values of the elements of  $\Gamma_b$  can be assumed to be equal. This simplification results in the hybrid equalizer tap-update 25 equations.

Performance degradation due to error propagation is often reduced by using better decision devices instead of a simple decision device. In particular, a trellis coded

modulated system can take advantage of the trellis decoder to obtain reliable decisions that can be fed back to the feedback path of the DFE.

Although various embodiments are specifically illustrated and described herein, it will be appreciated that modifications and variations of the invention are covered by the above teachings and are within the purview of the appended claims without departing from the spirit and intended scope of the invention. Furthermore, these examples should not be interpreted to limit the modifications and variations of the invention covered by the claims but are merely illustrative of possible variations.